Frequency Analysis of Temporal Graph Signals

Andreas Loukas* and Damien Foucard

Abstract—This letter extends the concept of graph-frequency to graph signals that evolve with time. Our goal is to generalize and, in fact, unify the familiar concepts from time- and graph-frequency analysis. To this end, we study a joint temporal and graph Fourier transform (JFT) and demonstrate its attractive properties. We build on our results to create filters which act on the joint (temporal and graph) frequency domain, and show how these can be used to perform interference cancellation. The proposed algorithms are distributed, have linear complexity, and can approximate any desired joint filtering objective.

I. INTRODUCTION

The recent availability of complex and high-dimensional datasets has spurred the need for new data analysis methods. One prominent research direction in signal processing has been the focus on data supported over graphs [1]. Graph signals, i.e., signals taking values on the nodes of combinatorial graphs, represent a convenient solution to model data exhibiting complex and non-uniform properties, such as those found in social, biological, and transportation networks, among others. Arguably, the most fundamental tool in the analysis of graph signals is the graph Fourier transform (GFT) [1]-[3]. In an analogous manner to the discrete Fourier transform (DFT), using GFT one may examine graph signals in the graph frequency domain, and, for instance, remove noise by attenuating high graph-frequencies. GFT has also lead to significant new insights in problems such as smoothing and denoising [4]–[6], segmentation [7], sampling and approximation [8]–[10], and classification [11]-[13] of graph data.

Yet, for many modern graph datasets, *time* is still of the essence. Whether we are interested in which candidate is more popular to whom in the political blogosphere [14], how an infection spreads over the global transportation network [15], or what the average daily traffic over the streets of a city is [16], the graph signals one encounters are not only a function of the underlying graph—they also evolve with time. Motivated by this need, this paper considers the frequency analysis of graph signals that change with time, referred to as *temporal graph signals*. Our goal is to generalize and, in fact, unify the familiar concepts from time- and graph-frequency analysis so as to jointly consider graph and temporal aspects of data.

To this end, we advocate for a joint temporal and graph Fourier transform (JFT) constructed by taking the graph Fourier and discrete Fourier transforms jointly. Though this idea is known [3], this paper brings forth new insights: (i) While previously defined only for the adjacency matrix, we show that a joint Fourier transform can be defined over a

The authors are with the Department of Telecommunication Systems, TU Berlin, Germany. e-mails: {a.loukas, d.foucard}@tu-berlin.de. *Corresponding author: A. Loukas.

wider selection of matrix representations of a graph, such as the discrete and normalized Laplacian. (ii) We study the properties and relations between Fourier transforms. This entails providing conditions for JFT to be unitary, as well as showing that JFT and GFT are equivalent under a certain transformation of the input graph, a relation which provides insight into the operation of JFT and demonstrates the consistency of our approach with the established theory. (iii) We propose a generalization of the notion of graph signal smoothness (or more precisely variation) appropriate for temporal graph signals.

We build on our results to design *joint filters*, which selectively attenuate or amplify certain joint-frequencies of a temporal graph signal. Contrary to previous work [3], [17], the proposed filters are distributed, have complexity linear in the number of graph edges, period of the signal and approximation order, and furthermore can approximate *any* desired joint-frequency response. In particular, we demonstrate how they can be used to approximate an interference cancellation problem, where given the statistical properties of a desired and interfering temporal graph signal, one is asked to design the filter which recovers the original signal with the smallest mean-squared error.

II. JOINT FOURIER TRANSFORM

A. A Transform for Temporal Graph Signals

Consider a graph $\mathcal{G}=(V,E)$ of N nodes u_1,\ldots,u_N and M edges and suppose that we are given a periodic temporal graph signal represented by a $\mathbb{R}^{N\times T}$ matrix \boldsymbol{X} , with X_{nt} being the value of node u_n at time instant t. Our goal is then to characterize the spectral properties of \boldsymbol{X} . Since each node has as values a temporal periodic signal, common wisdom dictates a transformation from the time to the frequency domain. Applying the discrete-Fourier transform on each row of \boldsymbol{X} , we obtain the frequency representation of our signal

$$DFT(X) = X\Psi_T^{\top}, \tag{1}$$

with the unitary matrix $\Psi_T = D/\sqrt{T}$ (i.e., $\Psi_T^* \times \Psi_T = I$) constructed as a normalization of the DFT matrix D of dimension $T \times T$. However, since the transform matrix Ψ_T^{\top} acts on each of the rows of X independently, it overlooks the graph structure of our data. Similarly, applying the graph Fourier transform in parallel [1], [3] for each time-instant as

$$GFT(X;\mathcal{G}) = \Psi_G X, \tag{2}$$

where Ψ_G is the $N \times N$ left eigenvector matrix of a matrix representation of \mathcal{G} , such as the Laplacian \mathcal{L}_G , normalized Laplacian N_G or the adjacency matrix A_G , lets us take into account the variation of the signal with respect to the graph, but neglects the temporal aspect of the data.

To capture the frequency content of X along both temporal and graph domains, one has to apply both transforms jointly. We can therefore define a joint graph and temporal Fourier transform as

$$JFT(X;\mathcal{G}) := \Psi_G X \Psi_T^\top. \tag{3}$$

In contrast to [3], the definition above is independent of the matrix representation of \mathcal{G} and can be used in conjuction with each definition of GFT. It might be more convenient to express JFT as a matrix vector multiplication. Exploiting the properties of the Kronecker product (\otimes) , we can write

$$JFT(x;\mathcal{G}) = (\Psi_T \otimes \Psi_G) x = \Psi_J x, \tag{4}$$

where in the last step we set $\Psi_J = \Psi_T \otimes \Psi_G$. Let us examine some properties of JFT.

Property 1. JFT is an invertible transform. The inverse transform in matrix and vector form is $JFT^{-1}(Y) = \Psi_G^{-1}Y\Psi_T^{T^*}$ and $JFT^{-1}(y) = (\Psi_T^* \otimes \Psi_G^{-1})y$, respectively, where y = vec(Y) = JFT(x). This can be confirmed using the fact that Ψ_T is unitary. For convenience of notation, in the following we set $\Phi_G = \Psi_G^{-1}$, $\Phi_T = \Psi_T^*$ and $\Phi_J = \Phi_T \otimes \Phi_G$, such that $DFT^{-1}(y) = \Phi_T y$, $GFT^{-1}(y) = \Phi_G y$, and $JFT^{-1}(y) = \Phi y$.

Property 2. JFT is a unitary transform if and only if GFT is unitary. JFT is a unitary transform when $\Psi_J \Psi_J^* = I_{NT}$. From definition, we have

$$\Psi_J \Psi_J^* = (\Psi_T \otimes \Psi_G) (\Psi_T \otimes \Psi_G)^*$$

$$= (\Psi_T \Psi_T^*) \otimes (\Psi_G \Psi_G^*) = I_T \otimes (\Psi_G \Psi_G^*). \tag{5}$$

For the last statement to be equal to I_{NT} (i.e., an identity matrix of dimension NT) it must be that $\Psi_G \Psi_G^* = I_N$, which is equivalent to asserting that GFT is unitary.

We deduce that JFT is a unitary transform for all symmetric matrix representations of a graph, such as the Laplacian or adjacency matrix, as long as the graph is undirected. On the other hand, when the graph is directed, unitarity is lost. It also follows that, if a symmetric matrix representation is used: (i) The columns of Φ_J form an orthonormal basis, and (ii) JFT obeys the Parseval theorem. For clarity, in the rest of this paper we opt to work with undirected graph $\mathcal G$ and only express our results w.r.t. the Laplacian matrix. Still, all results are directly applicable to alternative matrix representations.

Property 3. JFT is independent of the order DFT and GFT are applied on X. This is a direct consequence of the associativity of matrix multiplication.

B. Transform Equivalence

Suppose that we are given a graph \mathcal{G} of N nodes and a periodic temporal graph signal X of period T. Fig. 1 characterizes the relations between DFT, GFT, and JFT of X. Each directed arrow (e.g., from A to B) in the figure should be interpreted as a transform-simulation (transform A can be simulated by B). The equivalence between GFT and JFT is illustrated as a bidirectional simulation. Let us begin from the obvious relations. By definition, both DFT and GFT are

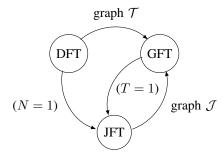


Fig. 1: Relations between Fourier transforms. Each directed arrow (say from A to B) in the figure should be interpreted as a transform-simulation (transform A can be simulated by B). Edge annotations hint on the simulation method.

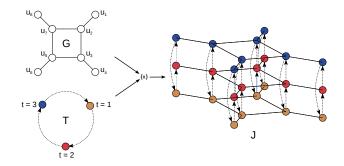


Fig. 2: The joint graph \mathcal{J} is the graph cartesian product of the input graph \mathcal{G} and a cycle graph \mathcal{T} .

specific cases of JFT. In particular, $DFT(X) = JFT(X; \mathcal{G})$ if the graph consists of a single node (N = 1) and $GFT(X; \mathcal{G}) = JFT(X; \mathcal{G})$ if X does not change in the temporal domain (T = 1).

We proceed with the remaining two relations.

 $DFT \rightarrow GFT$. To establish that DFT can be simulated using GFT, we will identify a graph $\mathcal{T} = (V_T, E_T)$ such that DFT(X) = GFT(X; T). The last relation is equivalent to requiring that the left and right eigenvector matrices of the Laplacian \mathcal{L}_T of graph T are Ψ_T and Φ_T , respectively. We obtain \mathcal{T} by thinking of (periodic) time as a ring graph consisting of T nodes, one per time-instant [18], [19]. In other words, each node $u_t \in V_T$ is connected to node u_{t+1} for $t = 1, \dots, T$, with index T + 1 = 1. The adjacency matrix of $\mathcal T$ is a circulant matrix and is known to have $oldsymbol{\Psi}_T$ and Φ_T as left and right eigenvector matrices, and as eigenvalues $\lambda_T(t) = \exp((2\pi i(t-1)(T-1))/T)$. Furthermore, since the Laplacian of \mathcal{T} is given by $\mathcal{L}_T = \mathbf{I}_T - \mathbf{A}_T$, it has the same eigenvectors and eigenvalues (up to translation and reordering), rendering the choice of representation (between A_T or \mathcal{L}_T or N_T) arbitrary.

JFT o GFT. We will simulate JFT by applying GFT on the joint graph $\mathcal{J} = (V_J, E_J)$, effectively showing that $JFT(X;\mathcal{G}) = GFT(X;\mathcal{J})$. We construct \mathcal{J} as the graph cartesian product of \mathcal{G} and \mathcal{T} . The joint graph consists of T copies of \mathcal{G} , denoted by $\mathcal{G}_t = (V_t, E_t)$, one for each timeinstant, with $V_J = V_1 \cup \ldots \cup V_T$. Name the corresponding nodes in each copy as $u_{n,t} \in V_t$. In addition to the $T \times M$ edges already introduced, the joint graph contains $T \times N$ extra edges joining consecutive copies: in particular, for each node $u_{n,t}$ in \mathcal{G}_t the joint graph has a directed edge to node $u_{n,t+1}$

in \mathcal{G}_{t+1} (modulo T). The Laplacian matrix of \mathcal{J} is expressed as

$$\mathbf{L}_{I} = \mathbf{I}_{T} \otimes \mathbf{L}_{G} + \mathbf{L}_{T} \otimes \mathbf{I}_{N} = \mathbf{L}_{T} \oplus \mathbf{L}_{G}, \tag{6}$$

where (\oplus) is the knonecker sum operator. Even though \mathcal{L}_J is not a symmetric matrix (due to \mathcal{T} and \mathcal{J} being directed) it follows from Theorem 13.16 in [20] that \mathcal{L}_J has eigendecomposition

$$\mathcal{L}_{J} = (\boldsymbol{\Phi}_{T} \otimes \boldsymbol{\Phi}_{G})(\boldsymbol{\Lambda}_{T} \oplus \boldsymbol{\Lambda}_{G})(\boldsymbol{\Psi}_{T} \otimes \boldsymbol{\Psi}_{G}) = \boldsymbol{\Phi}_{J}\boldsymbol{\Lambda}_{J}\boldsymbol{\Psi}_{J},$$

which fulfills out requirement.

C. A Joint Notion of Variation

The utility of a transform stems largely from its ability to provide insight about data. For instance, by observing the GFT of a graph signal one gains intuition about the *variation* of a signal over the graph, a notion which characterizes how aggressively a signal is changing on the graph. Therefore, GFT is useful because it allows us to distinguish smooth signals from non-smooth ones. In a similar manner, to render JFT a useful transform, beyond that of being a combination of two other transforms, we must give it insightful meaning.

We propose to use the relation $JFT(X;\mathcal{G}) = GFT(X;\mathcal{J})$ in order to imbue JFT with an appropriate notion of smoothness. In this way the variation of a temporal graph signal X on \mathcal{G} is defined to be equal to the variation of the same signal (interpreted now as a graph signal) on the joint graph \mathcal{J} .

Consider a node $u_{n_i,t} \in V_t \subseteq V_J$ and denote by $n_j \stackrel{\mathcal{J}}{\sim} n_i$ its neighbors in \mathcal{J} . Using the definition of variation [1] on \mathcal{J} , we define the *local variation* of the temporal graph signal \boldsymbol{X} at the n_i -th node at time t to be

$$\|\nabla_{n_i,t} \boldsymbol{X}\|_2 := \left[\sum_{n_j \lesssim n_i} \left(\frac{\partial \boldsymbol{X}}{\partial e_{n_i n_j}} \right)^2 \right]^{\frac{1}{2}}$$

$$= \left[\sum_{n_j \lesssim n_i} \left(X_{n_j,t} - X_{n_i,t} \right)^2 + \left(X_{n_i,t-1} - X_{n_i,t} \right)^2 \right]^{\frac{1}{2}}, \quad (7)$$

where $\frac{\partial \mathbf{X}}{\partial e_{n_i n_j}}$ is the discrete edge derivative on the joint graph and, in the last equation, $n_j \stackrel{\mathcal{G}}{\sim} n_i$ are the neighbors of u_{n_i} in \mathcal{G} . We can also obtain a global notion of smoothness using the p-Dirichlet form

$$S_p(\mathbf{X}) := \frac{1}{p} \sum_{n=1}^{N} \sum_{t=1}^{p} \|\nabla_{n,t} \mathbf{X}\|_2^p$$
 (8)

$$= \frac{1}{p} \sum_{n_i=1}^{N} \left[\sum_{\substack{n_j \in n_i}} (X_{n_j,t} - X_{n_i,t})^2 + (X_{n_i,t-1} - X_{n_i,t})^2 \right]^{\frac{p}{2}}.$$

For p = 2 and after some manipulation, we find that

$$S_2(\boldsymbol{X}) = \operatorname{vec}(\boldsymbol{X})^{\top} \boldsymbol{\mathsf{L}}_J \operatorname{vec}(\boldsymbol{X}) = \boldsymbol{x}^{\top} \boldsymbol{\mathsf{L}}_J \boldsymbol{x}. \tag{9}$$

¹The argument is identical for the adjacency and normalized Laplacian matrix representations.

Similarly to the GFT, $S_2(\boldsymbol{X})$ is a quadratic form of the (joint) Laplacian, which implies that $S_2(\boldsymbol{X}) \geq 0$. Yet, here the variation of a signal is not only w.r.t. \mathcal{G} but also w.r.t. time. For instance, $S_2(\boldsymbol{X}) = 0$ only if the signal is constant across all nodes and time-instances and, in general, the slower the values change along the graph and temporal domains, the smaller $S_2(\boldsymbol{X})$ becomes. Moreover, according to the Courant-Fischer theorem, the signals which minimize $S_2(\boldsymbol{X})$ are exactly the eigenvectors of L_J (i.e., the rows of $\boldsymbol{\Psi}_J$) with the corresponding minima being the associated eigenvalues λ_J of L_J [21]. JFT therefore characterizes a signal by how close its projections lie to the minimizers of the global variation $S_2(\boldsymbol{X})$; meaning that terms of low joint frequency λ_J (projections to eigenvectors associated with small eigenvalues) correspond to smoother signals and vice-versa.

III. JOINT FILTERS

To define the joint filtering problem in the most general form, we will consider a two-dimensional frequency domain, with the two dimensions conveying respectively the time- and graph- frequency of the signal. That is, one may now define joint frequency response $h(\lambda_T, \lambda_G)$ describing how the filter should change the frequency components independently w.r.t. λ_T and λ_G .

$$\mathbf{F}\mathbf{x} = \sum_{t=1,n=1}^{T,N} h(\lambda_T(t), \lambda_G(n)) \, \boldsymbol{\phi}_J(t,n) \boldsymbol{\psi}_J^*(t,n) \, \boldsymbol{x}$$
(10)

Above $\psi_J^*(t,n) = \psi_T^*(t) \otimes \psi_G^*(n)$ and $\phi_J(t,n) = \phi_T(t) \otimes \phi_G(n)$ are the eigenvectors giving eigenvalue $\lambda_T(t) + \lambda_G(n)$.

Joint filter design. In the spirit of FIR filters [5], our approach will be to first approximate the desired joint frequency response $h^*(\lambda_T, \lambda_G)$ by a bivariate polynomial of orders K, L in time and graph, respectively

$$h(\lambda_T, \lambda_G) = \sum_{k=0, \ell=0}^{K, L} c_{kl} \lambda_T^k \lambda_G^{\ell}, \tag{11}$$

with coefficients c_{kl} chosen to minimize a certain norm (such as the max-norm or the euclidean norm) of the approximation error. The corresponding joint filter is

$$\boldsymbol{F}\boldsymbol{x} = \sum_{k=0,\ell=0}^{K,L} c_{kl} \left(\boldsymbol{\mathsf{L}}_T^k \otimes \boldsymbol{\mathsf{L}}_G^\ell \right) \boldsymbol{x} \tag{12}$$

which, can be easily shown to possess the required response (by taking the eigenvalue decomposition of L_T and L_G , applying the mixed-product property of the Kronecker product, and exchanging the sums). Figure 3a illustrates the approximation error $\|h^* - h\|_2/\|h^*\|_2$ in the challenging case when the desired response is an ideal low-pass filter in time as well as graph frequency, $h^*(\lambda_T, \lambda_G) = 1$ if $\operatorname{angle}(\lambda_T) \leq \pi$ and $\lambda_G \leq 1$, and $h^*(\lambda_T, \lambda_G) = 0$, otherwise (note that since we use the normalized Laplacian, $0 \leq \lambda_G \leq 2$). Observe that the quality of approximation increases with the polynomial degrees, leading to a error of 0.54 for $K, L \geq 15$. Smaller errors can be achieved for continuous h^* functions (see for instance Fig. 3b).

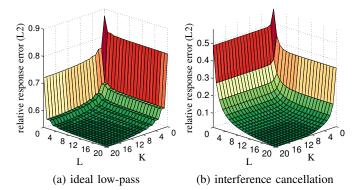


Fig. 3: Goodness of joint polynomial approximation of (a) an ideal joint low-pass filter and (b) a rational function used for interference cancellation, for varying approximation orders.

Computation. Because (12) involves powers of the input signal, it can be computed distributedly: each term $\mathbb{E}_T^k \otimes \mathbb{E}_G^\ell x$ is computed by iteratively multiplying the signal by $I_N \otimes \mathbb{E}_G$ (ℓ times) and $\mathbb{E}_T \otimes I_N$ (ℓ times), with each multiplication being a local operator on the joint graph and requiring the communication of 2MT and NT values, respectively. We can also reduce the overall complexity by computing terms $\mathbb{E}_T^k \otimes \mathbb{E}_G^\ell x$ recursively from either $\mathbb{E}_T^k \otimes \mathbb{E}_G^{\ell-1} x$ or $\mathbb{E}_T^{k-1} \otimes \mathbb{E}_G^\ell x$. Since $2MT \geq NT$, the most efficient scheme, which involves computing first all powers $I_T \otimes \mathbb{E}_G^\ell x$ and then using them to compute remaining terms, requires the exchange of 2MTK + (K+1)NTL = TK(2M+NL) + NTL = O(MTKL) values.

Interference cancellation. Suppose that we want to recover a graph signal $u \in \mathbb{R}^{NT}$ from an interfering signal $w \in \mathbb{R}^{NT}$. In this problem instance however, the two signals possess strong statistical structure in different domains. Let f and g be arbitrary matrix functions. We assume that, whereas u, which has covariance $\Sigma_u = I_T \otimes g(\mathbb{E}_G)$, has statistical properties that are *not* a function of time, the opposite holds for w with covariance $\Sigma_w = f(\mathbb{E}_T) \otimes I_N$, whose statistical properties are entirely temporal (this is a generalization of the models in [22]).

For simplicity, suppose that both signals are zero-mean. According to Wiener filter theory, the linear operator \bar{F} that recovers u from x=u+w with minimal mean-squared error is

$$\bar{F} = \underset{F}{\operatorname{argmin}} \mathbf{E} \left[\frac{\|Fx - u\|_2^2}{NT} \right] = \Sigma_u (\Sigma_u + \Sigma_w)^{\dagger},$$
 (13)

where using the pseudo-inverse (†) instead of the normal matrix inverse allows us to extend the result to positive semi-definite covariances. We then have that

$$(\boldsymbol{\Sigma}_{u} + \boldsymbol{\Sigma}_{w})^{\dagger} = (\boldsymbol{I}_{T} \otimes \boldsymbol{g}(\boldsymbol{\mathbb{E}}_{G}) + \boldsymbol{f}(\boldsymbol{\mathbb{E}}_{T}) \otimes \boldsymbol{I}_{N})^{\dagger}$$

$$= (\boldsymbol{f}(\boldsymbol{\mathbb{E}}_{T}) \oplus \boldsymbol{g}(\boldsymbol{\mathbb{E}}_{G}))^{\dagger}$$

$$= \boldsymbol{\Phi}_{J} (\boldsymbol{f}(\boldsymbol{\Lambda}_{T}) \oplus \boldsymbol{g}(\boldsymbol{\Lambda}_{G}))^{\dagger} \boldsymbol{\Psi}_{J}. \tag{14}$$

In addition.

$$\Sigma_{u} = \mathbf{I}_{T} \otimes \mathbf{g}(\mathbb{L}_{G}) = \mathbf{\Phi}_{T} \mathbf{\Psi}_{T} \otimes \mathbf{\Phi}_{G} \mathbf{g}(\mathbf{\Lambda}_{G}) \mathbf{\Psi}_{G}
= (\mathbf{\Phi}_{T} \otimes \mathbf{\Phi}_{G}) (\mathbf{I}_{T} \otimes \mathbf{g}(\mathbf{\Lambda}_{G})) (\mathbf{\Psi}_{T} \otimes \mathbf{\Psi}_{G})
= \mathbf{\Phi}_{J} (\mathbf{I}_{T} \otimes \mathbf{g}(\mathbf{\Lambda}_{G})) \mathbf{\Psi}_{J}.$$
(15)

Since $\Psi_I \Phi_J = I_{NT}$, we conclude that

$$\bar{\mathbf{F}} = \mathbf{\Phi}_J \left(\mathbf{I}_T \otimes \mathbf{g}(\mathbf{\Lambda}_G) \right) \left(\mathbf{f}(\mathbf{\Lambda}_T) \oplus \mathbf{g}(\mathbf{\Lambda}_G) \right)^{\dagger} \mathbf{\Psi}_J, \tag{16}$$

which is a joint filter with response $h(\lambda_T,\lambda_G)=g(\lambda_G)/(g(\lambda_G)+f(\lambda_T))$ if $g(\lambda_G)+f(\lambda_T)\neq 0$ and $h(\lambda_T,\lambda_G)=0$, otherwise. Therefore, signals ${\boldsymbol u}$ and ${\boldsymbol w}$ cannot be well separated by acting disjointly on their respective domains; the best linear estimator $\bar{{\boldsymbol F}}{\boldsymbol x}$ of ${\boldsymbol u}$, is given by a filter acting on the joint Fourier domain. Moreover, as shown in Fig. 3b (now for the simple case of $f(\lambda_T)=\lambda_T+1$ and $g(\lambda_G)=\lambda_G$) operator $\bar{{\boldsymbol F}}$ can be tightly approximated by our proposed joint filters (12), even using moderate polynomial orders.

REFERENCES

- [1] D. Shuman, S. K. Narang, P. Frossard, A. Ortega, P. Vandergheynst et al., "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," Signal Processing Magazine, IEEE, vol. 30, no. 3, pp. 83–98, 2013.
- [2] D. Shuman, B. Ricaud, P. Vandergheynst et al., "A windowed graph fourier transform," in *Statistical Signal Processing Workshop (SSP)*, 2012 IEEE. Ieee, 2012, pp. 133–136.
- [3] A. Sandryhaila and J. M. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," Signal Processing Magazine, IEEE, vol. 31, no. 5, pp. 80–90, 2014.
- [4] F. Zhang and E. R. Hancock, "Graph spectral image smoothing using the heat kernel," *Pattern Recognition*, vol. 41, no. 11, pp. 3328–3342, 2008.
- [5] D. I. Shuman, P. Vandergheynst, and P. Frossard, "Chebyshev polynomial approximation for distributed signal processing," in *International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS)*. IEEE, 2011, pp. 1–8.
- [6] A. Loukas, M. A. Zúñiga, M. Woehrle, M. Cattani, and K. Langendoen, "Think Globally, Act Locally: On the Reshaping of Information Landscapes," in *International Conference on Information Processing in Sensor Networks (IPSN)*. ACM/IEEE, 2013.
- [7] A. Loukas, M. A. Zúñiga, I. Protonotarios, and J. Gao, "How to identify global trends from local decisions? Event Region Detection on Mobile Networks," in *International Conference on Computer Communications* (INFOCOM), 2014.
- [8] X. Zhu and M. Rabbat, "Graph spectral compressed sensing for sensor networks," in Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on. IEEE, 2012, pp. 2865–2868.
- [9] —, "Approximating signals supported on graphs." in *ICASSP*. Cite-seer, 2012, pp. 3921–3924.
- [10] S. K. Narang, A. Gadde, and A. Ortega, "Signal processing techniques for interpolation in graph structured data," in Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, 2013, pp. 5445–5449.
- [11] A. J. Smola and R. Kondor, "Kernels and regularization on graphs," in Learning theory and kernel machines. Springer, 2003, pp. 144–158.
- [12] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proceedings of the 20th International Conference on Machine Learning (ICML-2003) Volume* 2, vol. 2. AIAA Press, 2003, pp. 912–919.
- [13] M. Belkin and P. Niyogi, "Semi-supervised learning on riemannian manifolds," *Machine learning*, vol. 56, no. 1-3, pp. 209–239, 2004.
- [14] L. A. Adamic and N. Glance, "The political blogosphere and the 2004 us election: divided they blog," in *Proceedings of the 3rd international* workshop on Link discovery. ACM, 2005, pp. 36–43.
- [15] A. J. Tatem, D. J. Rogers, and S. Hay, "Global transport networks and infectious disease spread," *Advances in parasitology*, vol. 62, pp. 293– 343, 2006.

- [16] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones," in *Proceedings of the 6th ACM conference on Embedded network sensor systems*. ACM, 2008, pp. 323–336.
- [17] A. Loukas, A. Simonetto, and G. Leus, "Distributed Autoregressive Moving Average Graph Filters," Signal Processing Letters, vol. 22, no. 11, pp. 1931–1935, 2015.
- [18] M. Püschel and J. M. Moura, "Algebraic signal processing theory: 1-d space," Signal Processing, IEEE Transactions on, vol. 56, no. 8, pp. 3586–3599, 2008.
- [19] A. Sandryhaila and J. M. Moura, "Discrete signal processing on graphs," Signal Processing, IEEE Transactions on, vol. 61, no. 7, pp. 1644–1656, 2013
- [20] A. J. Laub, Matrix analysis for scientists and engineers. Siam, 2005.
- [21] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.
- [22] C. Zhang, D. Florencio, and P. Chou, "Graph signal processing a probabilistic framework," Tech. Rep. MSR-TR-2015-31, April 2015. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=243326